Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer-implemented method for adjusting the color information of an image by modelling a non-linear transfer function with a power law function, the method comprising:

receiving a transfer function, wherein the transfer function specifies a set of output values corresponding to a set of input values; ~~and~~

iteratively, until a termination flag is set:

receiving a first power law function;

generating an auxiliary function from the transfer function and local differences between the transfer function and the first power law function;

fitting a second power law function to the auxiliary function;

calculating a modelling error from the second power law function and the transfer function; and

setting the termination flag when the modelling error is less than a predetermined value; and

using the second power law function to calculate a gamma value that is used to adjust the color information of the image.

2. (Currently Amended) The method of claim ~~1~~ 3, wherein[[:]] in an iteration other than the first iteration, ~~receiving~~ the first power law function ~~in a given iteration comprises receiving~~ is the second power law function that was ~~generated~~ fit to the auxiliary function in the immediately preceding iteration.

3. (Currently Amended) The method of claim 1, wherein[[:]] in a first iteration, ~~receiving~~ the first power law function ~~in the first iteration comprises receiving~~ is a power law function obtained ~~generated~~ by fitting the transfer function.

4. (Original) The method of claim 1, further comprising:

counting the number of iterations; and

setting the termination flag when the number of iterations exceeds a maximum number of iterations.

5. (Original) The method of claim 1, wherein:

the transfer function is a transfer function for gamma correction, and the first and second power law functions are power law functions having a form of cx^{β}, wherein x is the input variable of the power law functions, and c and β are real numbers.

6. (Original) The method of claim 5, wherein:

fitting the second power law function to the auxiliary function includes fitting a linear function to a logarithmic representation of the auxiliary function.

7. (Original) The method of claim 6, wherein:

fitting the linear function to the logarithmic representation of the auxiliary function includes minimizing a least square error between the linear function and the logarithmic representation of the auxiliary function.

8. (Currently Amended) The method of claim 1, further comprising:

using a modifying parameter to weight the local differences between the transfer function and the first power law function, and using the weighted local differences ~~that are used~~ to generate the auxiliary function from the transfer function.

9. (Currently Amended) The method of claim 8, further comprising:

~~optimizing the modifying parameter~~ determining a value for the modifying parameter that minimizes the modelling error between the transfer function and the power

law function that is fit to the auxiliary function that is generated with that modifying parameter.

10. (Currently Amended) The method of claim 9, wherein ~~optimizing~~ determining the value of the modifying parameter that minimizes the modelling error comprises :

generating a plurality of auxiliary functions from the transfer function, the first power law function and a corresponding plurality of modifying parameters, wherein each auxiliary function is generated by weighting the local differences between the transfer function and the first power law function using a corresponding one of the plurality of modifying parameters;

fitting each ~~auxiliary function in~~ of the plurality of auxiliary functions to generate a respective plurality of second power law functions, where each of the plurality of second power law functions corresponds to one of the plurality of modifying parameters;

calculating a plurality of modelling errors between the transfer function and each of the ~~for each power law function in the~~ plurality of second power law functions, wherein each of the modeling errors corresponds to one of the plurality of modifying parameters ~~to generate a plurality of modelling errors~~; and

~~determining an optimal modifying parameter from the plurality of modelling errors~~ executing a minimization procedure to determine the value of the modifying parameter that minimizes the modelling error.

11. (Currently Amended) The method of claim 10, wherein[[:]] ~~determining the optimal modifying parameter comprises determining a range of modifying parameters that includes the optimal modifying parameter~~ executing the minimization procedure comprises fitting a quadratic function to a distribution of modelling errors as a function of the plurality of modifying parameters.

12. (Currently Amended) The method of claim 10, wherein[[:]] ~~determining the optimal modifying parameter comprises performing~~ executing the minimization procedure

comprises executing a golden search algorithm ~~modelling error that corresponds to the optimal modifying parameter~~.

13. (Original) The method of claim 1, wherein: calculating the modelling error for the second power law function comprises calculating a total square error between the transfer function and the second power law function.

14. (Original) The method of claim 1, wherein: calculating the modelling error for the second power law function comprises calculating the maximum absolute difference between the transfer function and the second power law function.

15. (Original) The method of claim 1, wherein: receiving a transfer function comprises receiving a plurality of transfer function values.

16. (Currently Amended) The method of claim 1, wherein: receiving a transfer function comprises receiving a piecewise continuous monotonically increasing ~~transfer~~ function.

17. (Currently Amended) A computer-implemented method for adjusting the color information of an image by modelling a non-linear transfer function with a power law function, the method comprising:

receiving a transfer function, wherein the transfer function specifies a set of output values corresponding to a set of input values;

fitting the transfer function with ~~an approximating~~ a first power law function; ~~and~~

iteratively, until a termination flag is set:

reflecting the ~~approximating~~ first power law function about the transfer function to generate an auxiliary function;

fitting the auxiliary function with a ~~new approximating~~ second power law function;

calculating a modelling error from the ~~new approximating~~ second power law function and the transfer function; ~~and~~

identifying the first power law function with the second power law

function; and

setting the termination flag when the modelling error is less than a predetermined value; and

using the second power law function to calculate a gamma value that is used to adjust the color information of the image.

18. (Currently Amended) A ~~software~~ computer program product, ~~tangibly embodied in an information carrier~~ implemented on a machine readable medium, for adjusting the color information of an image by modelling a non-linear transfer function with a power law function, the ~~software~~ computer program product comprising instructions operable to cause a programmable processor to ~~one or more data processing apparatus to perform operations comprising~~:

~~receiving~~ receive a transfer function, wherein the transfer function specifies a set of output values corresponding to a set of input values; ~~and~~

iteratively, until a termination flag is set:

~~receiving~~ receive a first power law function;

~~generating~~ generate an auxiliary function from the transfer function and local differences between the transfer function and the first power law function;

~~fitting~~ fit a second power law function to the auxiliary function;

~~calculating~~ calculate a modelling error from the second power law function and the transfer function; and

~~setting~~ set the termination flag when the modelling error is less than a predetermined value; and

use the second power law function to calculate a gamma value that is used to adjust the color information of the image.

19. (Currently Amended) The ~~software~~ computer program product of claim ~~18~~ 20, wherein[[:]] in an iteration other than the first iteration, ~~receiving~~ the first power law

function ~~in a given iteration comprises receiving~~ is the second power law function that was ~~generated~~ fit to the auxiliary function in the immediately preceding iteration.

20. (Currently Amended) The ~~software~~ computer program product of claim 18, wherein[[:]] in a first iteration, ~~receiving~~ the first power law function ~~in the first iteration comprises receiving a power law function~~ is obtained ~~generated~~ by fitting the transfer function.

21. (Currently Amended) The ~~software~~ computer program product of claim 18, further comprising instructions operable to cause the programmable processor to ~~one or more data processing apparatus to perform operations comprising~~:

~~counting~~ count the number of iterations; and to

~~setting~~ set the termination flag when the number of iterations exceeds a maximum number of iterations.

22. (Currently Amended) The ~~software~~ computer program product of claim 18, wherein:

the transfer function is a transfer function for gamma correction, and the first and second power law functions are power law functions having a form of cx^{β}, wherein x is the input variable of the power law functions, and c and β are real numbers.

23. (Currently Amended) The ~~software~~ computer program product of claim 22, wherein [[:]] the instructions to ~~fitting~~ fit the second power law function to the auxiliary function includes instructions to ~~fitting~~ fit a linear function to a logarithmic representation of the auxiliary function.

24. (Currently Amended) The ~~software~~ computer program product of claim 23, wherein [[:]] the instructions to ~~fitting~~ fit the linear function to the logarithmic representation of the auxiliary function includes instructions to ~~minimizing~~ minimize a least square error between the linear function and the logarithmic representation of the auxiliary function.

25. (Currently Amended) The ~~software~~ computer program product of claim 18, further comprising instructions operable to cause the programmable processor to ~~one or more~~

~~data processing apparatus to perform operations comprising~~:

~~using~~ use a modifying parameter to weight the local differences between the transfer function and the first power law function, and to use the weighted local differences ~~that are used~~ to generate the auxiliary function from the transfer function.

26. (Currently Amended) The ~~software~~ computer program product of claim 25, further comprising instructions operable to cause the programmable processor to ~~one or more data processing apparatus to perform operations comprising:optimizing the modifying parameter~~ determine a value for the modifying parameter that minimizes the modelling error between the transfer function and the power law function that is fit to the auxiliary function that is generated with that modifying parameter.

27. (Currently Amended) The ~~software~~ computer program product of claim 26, wherein the instructions to ~~optimizing the~~ determine the value of the modifying parameter that minimizes the modelling error comprises instructions to:

~~generating~~ generate a plurality of auxiliary functions from the transfer function, the first power law function and a corresponding plurality of modifying parameters, wherein each auxiliary function is generated by weighting the local differences between the transfer function and the first power law function using a corresponding one of the plurality of modifying parameters;

~~fitting each auxiliary function in~~ fit each of the plurality of auxiliary functions to generate a respective plurality of second power law functions, where each of the plurality of second power law functions corresponds to one of the plurality of modifying parameters;

~~calculating a~~ calculate a plurality of modelling errors between the transfer function and each of the ~~for each power law function in the~~ plurality of second power law functions, wherein each of the modelling errors corresponds to one of the plurality of modifying parameters ~~to generate a plurality of modelling errors~~; and

~~determining an optimal modifying parameter from the plurality of~~ modelling

~~errors~~ execute a minimization procedure to determine the value of the modifying parameter that minimizes the modelling error.

28. (Currently Amended) The ~~software~~ computer program product of claim 27, wherein[[:]] ~~determining the optimal modifying parameter comprises determining a range of modifying parameters that includes the optimal modifying parameter~~ the instructions to execute the minimization procedure comprise instructions to fit a quadratic function to a distribution of modelling errors as a function of the plurality of modifying parameters.

29. (Currently Amended) The ~~software~~ computer program product of claim 27, wherein[[:]] ~~determining the optimal modifying parameter comprises performing~~ the instructions to execute the minimization procedure comprise instructions to execute a golden search algorithm ~~for the modelling error that corresponds to the optimal modifying parameter~~.

30. (Currently Amended)The ~~software~~ computer program product of claim 18, wherein [[:]] the instructions to calculate ~~calculating~~ the modelling error for the second power law function comprises ~~calculating~~ instructions to calculate a total square error between the transfer function and the second power law function.

31. (Currently Amended) The ~~software~~ computer program product of claim 18, wherein [[:]]the instructions to calculate ~~calculating~~ the modelling error for the second power law function comprises ~~calculating~~ instructions to calculate the maximum absolute difference between the transfer function and the second power law function.

32. (Currently Amended) The ~~software~~ computer program product of claim 18, wherein [[:]] the ~~receiving a~~ received transfer function comprises ~~receiving~~ a plurality of transfer function values.

33. (Currently Amended) The ~~software~~ computer program product of claim 18, wherein [[:]] the received ~~receiving a~~ transfer function comprises ~~receiving~~ a piecewise continuous monotonically increasing ~~transfer~~ function.

34. (Currently Amended) A ~~software~~ computer program product, ~~tangibly embodied in an information carrier~~ implemented on a machine readable medium, for adjusting the color information of an image by modelling a non-linear transfer function with a power law function, the ~~software~~ computer program product comprising instructions operable to cause ~~one or more data processing apparatus to perform operations comprising~~ a programmable processor to:

~~receiving~~ receive a transfer function, wherein the transfer function specifies a set of output values corresponding to a set of input values;

~~fitting~~ fit the transfer function with ~~an approximating~~ a first power law function; and to iteratively, until a termination flag is set:

~~reflecting~~ reflect the ~~approximating~~ first power law function about the transfer function to generate an auxiliary function;

~~fitting~~ fit the auxiliary function with a ~~new approximating~~ second power law function;

~~calculating~~ calculate a modelling error from the ~~new approximating~~ second power law function and the transfer function; ~~and~~

identify the first power law function with the second power law function; and

~~setting~~ set the termination flag when the modelling error is less than a predetermined value; and

use the second power law function to calculate a gamma value that is used to adjust the color information of the image.